

Fonaments de la programació

Arrays i el bucle For

`{.js}`
JavaScript



UNIÓ EUROPEA
Fons Europeu de
Desenvolupament Regional

Estructures en JavaScript

Estructures condicionals

Operador condicional
?:

if

if-else

if niats

switch

Salts

break

M4 ✓

Estructures en bucle

while

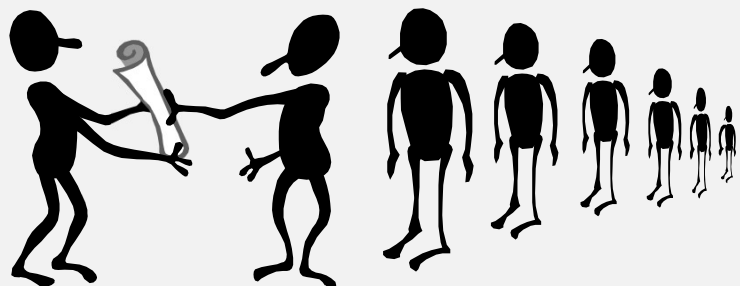
do - while

for

M5

1

Arrays



Arrays

Un **vector** o **array** és una col·lecció finita de dades **del mateix tipus**, que s'emmagatzemen en posicions consecutives de memòria, **reben un nom comú** i **permet a més la utilització individual dels seus elements**.

És important destacar que la **primera posició d'un array sempre és la posició 0**.

Exemple: `alumnes`

Pep	Paco	Lluïsa	Aina	Joan
<code>alumnes[0]</code>	<code>alumnes[1]</code>	<code>alumnes[2]</code>	<code>alumnes[3]</code>	<code>alumnes[4]</code>

Utilitzar Arrays

Crearem el array `alumnes` de la diapositiva 4

```
let alumnes = [];  
  
alumnes.push("Pep");  
alumnes.push("Paco");  
alumnes.push("Aina");  
alumnes.push("Lluïsa");  
alumnes.push("Joan");
```

```
alert(alumnes);
```

```
alert(alumnes[2]);
```

A JavaScript, un array pot emmagatzemar elements de **qualsevol tipus**.

Creem l'array i amb el mètode `push` l'omplim amb els elements

[Pep, Paco, Aina, Lluïsa, Joan]

Aina

Operacions amb Arrays

Pep	Paco	Lluïsa	Aina	Joan
<code>alumnes[0]</code>	<code>alumnes[1]</code>	<code>alumnes[2]</code>	<code>alumnes[3]</code>	<code>alumnes[4]</code>

Operació	Métode	Exemple ⓘ
Afegir un element al darrer d'un array	<code>array.<u>push</u>(valor);</code>	<code>alumnes.push("Pep");</code>
Afegir o eliminar elements en un array en una posició donada. Si #elements són 0, afegiu els valors del contrari, suprimiu #elements d'aquesta posició i afegiu els valors donats.	<code>array.<u>splice</u>(posicio, #elements, valor1, valor2...)</code>	<pre>alumnes.splice(2,0,"Laura"); alert(alumnes); //Mostra Pep,Paco,Laura,Lluïsa,Aina,Joan alumnes.splice(2,1,"Laura"); alert(alumnes); //Mostra Pep,Paco,Laura,Aina,Joan</pre>
Reemplaçar un element	<code>array[posicio] = valor;</code>	<pre>alumnes[1]="Jordi"; //Canvia al Paco per al Jordi</pre>
Accedir a un element	<code>valor = array[posicio];</code>	<pre>let nen = alumnes[2]; alert(nen); // Mostra Lluïsa</pre>
Obtenir la mida d'un array	<code>array.<u>length</u>;</code>	<code>alert alumnes.length; //Mostra 5</code>

Ordenar els elements d'un array

```
let alumnes = [];  
  
alumnes.push("Pep");  
alumnes.push("Paco");  
alumnes.push("Aina");  
alumnes.push("Lluïsa");  
alumnes.push("Joan");  
  
alert(alumnes);  
  
alumnes.sort();  
alert(alumnes);  
  
alumnes.reverse();  
alert(alumnes);
```

La funció **sort()** ordena els elements com a cadenes en ordre alfabètic i ascendent mentre que la funció **reverse()** ordena els elements com a cadenes en ordre alfabètic i descendent.

[Pep, Paco, Aina, Lluïsa, Joan]

[Aina, Joan, Lluïsa, Paco, Pep]

[Pep, Paco, Lluïsa, Joan, Aina]

L'ordenació de números pot produir resultats incorrectes.

https://www.w3schools.com/jsref/jsref_sort.asp

{.js}

Fonaments de la programació **JavaScript**

Iterar sobre un Array

```
let alumnes = ['Pep', 'Paco', 'Aina', 'Lluisa', 'Joan'];
let contenido = "Contenido del Array alumnes:\n";
let i=0;

//Iterar

for (let valor of alumnes) {
    contenido += i + " " + valor + '\n';
    i++;
};

alert(contenido);
```

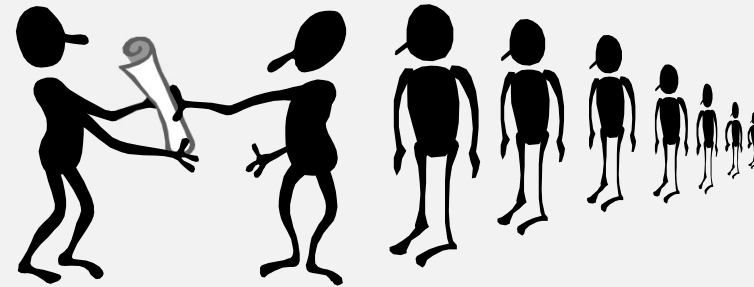
La sentència **for...of** executa un bloc de codi per a cada element d'un objecte iterable, com ho són: String, Arrays, i altres objectes com ara arguments TypedArray, Map, Set etc.

Esta página dice

Contenido del Array alumnes:

0 Pep
1 Paco
2 Aina
3 Lluisa
4 Joan

Aceptar



```
per (alumne = 1 ; alumne < 10 ; pròxim alumne) {  
    Lliurar DIPLOMA  
}
```

2

El bucle for

El bucle for

El bucle **for** permet repetir un nombre de vegades determinat un bloc de codi, per la qual cosa és ideal per a recórrer arrays.

El bucle for **s'utilitza quan sabem el número d'iteracions que el bucle s'ha d'executar.**

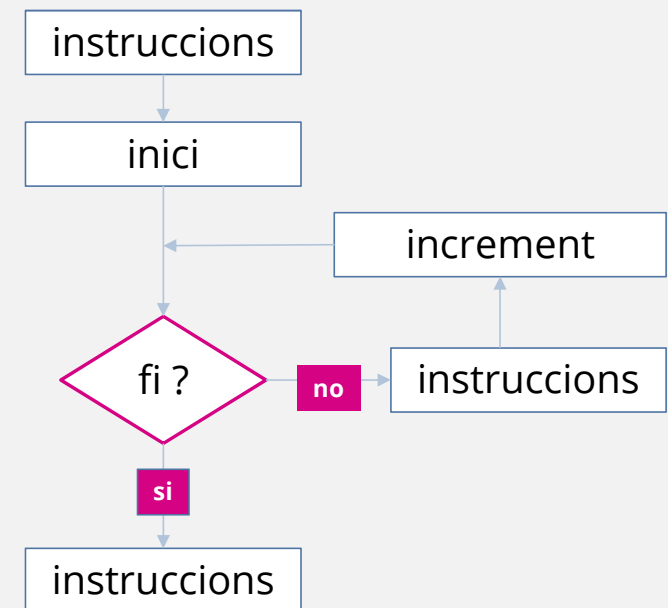
El bucle for **sempre ha de tenir un inici, un final i un increment.**

Instruccions...

```
for ( inici; final; increment ){  
    instrucció3;  
    instrucció4;  
}
```

sintaxi

Instruccions...



El bucle for

Exemple:

```
let alumnes = ['Pep', 'Paco', 'Aina', 'Lluisa', 'Joan'];

let contenido = "Contenido del Array alumnes:(bucle For):\n";
let i=0;

//Iterar

for (i=0; i<alumnes.length; i++) {
    contenido += i + " " + alumnes[i] + '\n';
};

alert(contenido);
```

Esta página dice

Contenido del Array alumnes (bucle For):

0 Pep
1 Paco
2 Aina
3 Lluisa
4 Joan

Aceptar

{.js}
JavaScript

Fonaments de la programació

Bucle for



Ordre execució

```
let lletres = ['a', 'b', 'c'];
for (let i = 0 ; i < lletres.length; i++) {
    alert(lletres[i]);
}
...
```



```
for (inicialització del comptador ; condició del bucle ; increment (o decrement) del comptador){
    Codi a executar
};
```

inicialització del comptador	<code>int i = 0;</code>	<code>i = 0</code>
condició del bucle	<code>i < lletres.length;</code> ►	<code>true</code>
Codi a executar	<code>alert(lletres[i]);</code>	<code>a</code>
increment del comptador	<code>i++</code>	<code>i = 1</code>
condició del bucle	<code>i < lletres.length;</code> ►	<code>true</code>
Codi a executar	<code>alert(lletres[i]);</code>	<code>b</code>
increment del comptador	<code>i++</code>	<code>i = 2</code>
condició del bucle	<code>i < lletres.length;</code> ►	<code>true</code>
Codi a executar	<code>alert(lletres[i]);</code>	<code>c</code>
increment del comptador	<code>i++</code>	<code>i = 3</code>
condició del bucle	<code>i < lletres.length;</code> ►	<code>false</code>